Retrieval of Web Documents Using a Fuzzy Hierarchical Clustering

Deepti Gupta
Lecturer
School of Computer Science and Information Technology
Shobhit University, Meerut, INDIA

Nidhi Tyagi
Asst. Professor
School of Computer Science and Information Technology
Shobhit University Meerut, INDIA

Komal Kumar Bhatia
Asst. Professor
Department of Computer Engineering
YMCA University of Science, Faridabad, INDIA

ABSTRACT

The World Wide Web has huge amount of information that is retrieved using information retrieval tool like Search Engine. Page repository of Search Engine contains the web documents downloaded by the crawler. This repository contains variety of web documents from different domains. In this paper, a technique called "Retrieval of Web documents using a fuzzy hierarchical clustering" is being proposed that creates the clusters of web documents using fuzzy hierarchical clustering.

Keywords

Search Engine, Web documents, Fuzzy Hierarchical Clustering

1. Introduction

WWW [01, 02 and 04] is a huge repository of information consisting of hyperlinked documents spread over the internet. The size of the web as on February 2010 stands at around 35 million pages. For a user, it is practically impossible to search through this extremely large database for the information needed by him. Hence the need for Search Engine (see fig. 1) arises. The search engine uses crawlers to gather information and stores it in database maintained at search engine side. For a given user's query the search engine searches in the local database and very quickly displays the results. The huge amount of information [05] is retrieved using data mining tools. Classification, Clustering and Association tools etc. are used for data mining technique. Clustering plays a key role in searching for structures in data. As the number of available documents nowadays is large, hierarchical approaches are better suited because they permit categories to be defined at different pensiveness levels. The problem of clustering in finite set of data is to find several cluster centers that can properly characterize relevant classes of finite set of data such that degree of association is strong for data within blocks of the partition and weak for data in different blocks. When the weakness of a crisp partition of finite set of data is replaced with a fuzzy partition, this area is known as fuzzy clustering.

Fuzzy clustering is a relevant technique for information retrieval. As a document might be relevant to multiple queries, this document should be given in the corresponding response sets, otherwise, the users would not be aware of it. Fuzzy clustering seems a natural technique for document categorization. There are two basic methods of fuzzy clustering, one which is based on fuzzy c-partitions, is called a fuzzy c-means clustering method and the other, based on the fuzzy equivalence relations, is called a fuzzy equivalence clustering method.

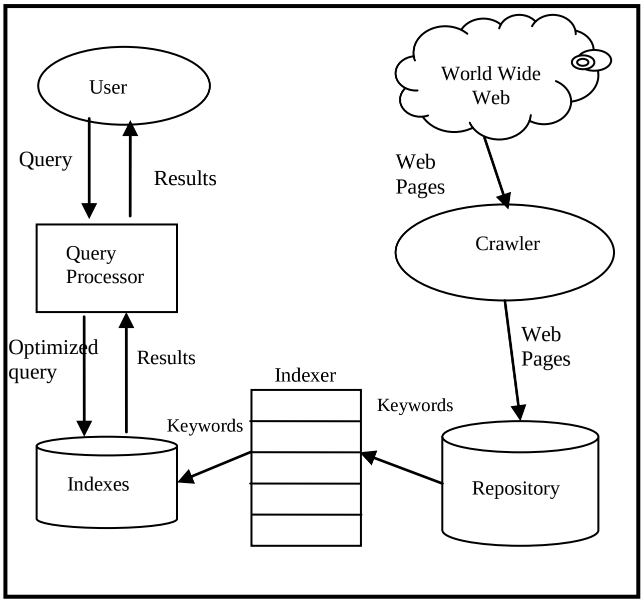

Fig. 1. General architecture of the Search Engine

The purpose of this research is to propose a search methodology that consists of how to find relevant information from WWW. In this paper, a method is being proposed of document clustering, which is based on fuzzy equivalence relation that helps information retrieval in the terms of time and relevant information.

The paper is structured as follows: section 2 describes some related work about fuzzy hierarchical clustering algorithms. Section 3 shows the proposed method and section 4 presents an example, how to retrieve the relevant information from WWW. Section 5 shows the results. In section 6, conclusion and future work are presented.

2. Related Work

The goal of document clustering is to categorize the documents so that all the documents in a cluster are similar. Most of the early work [03, 10] applied traditional clustering algorithms like K-means, to the set of documents to be clustered. Willett [11], provided a survey on applying hierarchical clustering algorithms into clustering documents.

Cutting et al. [07], proposed speeding up the partition based clustering by using techniques that provide good initial clusters. Two techniques, Buckshot and Fractionation are mentioned. Buckshot selects a small sample of documents to pre-cluster them using a standard clustering algorithm and assigns the rest of the documents to the clusters formed. Fractionation, splits the N documents into 'm' buckets where each bucket contains N/m documents. Fractionation takes an input parameter r, which indicates the reduction factor for each bucket. The standard clustering algorithm is applied so that if there are 'n' documents in each bucket, they are clustered into n/r clusters. Now each of these clusters are treated as if they were individual documents and the whole process is repeated until there are only 'K' clusters.

Torra et. al. [12] presents a detailed study of applying fuzzy hierarchical clustering algorithms in an extension of the Gambal system for clustering and visualization of documents. C-Means Fuzzy Hierarchical Clustering [13, 14 and 15] algorithms have mostly been used. It is a well-known method that generalizes the crisp clustering algorithm k-means so that partial membership is allowed.

According to Lefever et. al. [08], the challenging aspect of this task is that it is in general not known beforehand how many clusters to expect, therefore he proposed the use of a Fuzzy Ants clustering algorithm that does not rely on prior knowledge of the number of clusters that need to be found in the data. An evaluation on benchmark data sets from SemEval's WePS1 and WePS2 competitions shows that the resulting system is competitive with the agglomerative clustering Agnes algorithm. This is particularly interesting as the latter involves manual setting of a similarity threshold (or estimating the number of clusters in advance) while the former does not.

The critical look at the available literature reveals that, the hierarchical fuzzy c- means clustering technique [06, 09] and many more, have been implemented for the retrieval of the documents. This paper proposes a new method for the retrieval, the fuzzy equivalence relation. Fuzzy equivalence relation [10] is a hierarchical, bottom-up approach, where the relation between the documents if generated and more accurate clusters are formed. This method is more efficient in terms of time and accuracy.

3. Proposed Work

A clustering method based upon fuzzy equivalence relations is being proposed for information retrieval. The downloaded documents and the keywords contained therein and stored in a repository by the crawler see figure 1.1. The indexer extracts all words from the entire set of documents and eliminates non-content-bearing words i.e. stop words such as "a", "and", "the" etc from each documents. These keywords fetch the related documents and stored in the indexed database. The documents are stored in indexed database based on keywords. Now, the proposed fuzzy clustering method based upon fuzzy equivalence relations is applied on the indexed database. A list of common words called keywords is generated in table 3.1.

Table 3.1: Document No. and Keywords

Document No.	**Keywords**
0	Crawler
1	Search Engine, Database
2	Web
3	Search Engine
4	Crawler
5	Web

Each keyword is assigned a Keyword ID as shown in table 3.2.

Table 3.2: Keywords and Keywords ID

Keywords	**Keywords ID**
Crawler	0
Search Engine	1
Web	2
Database	3

The information contained in table 3.1 and table 3.2 is used to generate the required document clustering for applying fuzzy equivalence relation.

Since it is not directly possible; so first determine a fuzzy compatibility relation (reflexive and symmetric) in terms of an appropriate distance function applied on given data. Then, a meaningful fuzzy equivalence relation is defined as the transitive closure of this fuzzy compatibility relation.

A set of data X is consisting of the following six points in R2(p-tuples of Rp) as shown in figure 3.1.

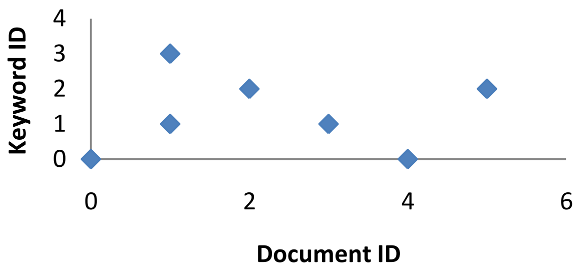

Fig. 3.1 : A Graphical representation between Document ID and Keyword ID

The data X is shown in table 3.3.

Table 3.3

K	**1**	**2**		**3**	**4**	**5**	**6**
x_{k1}	0	1	1	2	3	4	5
x_{k2}	0	1	3	2	1	0	2

Let a fuzzy compatibility relation, R, on X be defined in terms of an appropriate distance function of the Minkowski class by the formula

$$R(x_i,x_k) = 1- \delta(\sum_{J=1}^{p} | x_{ij} – x_{kj} |^q)^{1/q}.......(i)$$

For all pairs (xi,xk) Є X, where q Є RT, and δ is a constant that ensures that R(xi,xk) Є [0,1], Clearly , δ is the inverse value of the largest distance in X. In general, R defined by equation (i) is a fuzzy compatibility relation, but not necessarily a fuzzy equivalence relation. Hence, there is need to determine the transitive closure of R.

Given a relation R(X,X), its transitive closure RT (X,X) can be determined by simple algorithm that consists of the following three steps:

1. R' = R U (R o R)
2. If R' ≠ R, make R = R' and go to step 1
3. Stop R' = R_T

This algorithm is applicable to both crisp and fuzzy relations. However, the type of composition and set union in stepI must be compatibility with the definition of transitivity employed. After applying this algorithm a hierarchical cluster tree will be generated. Each cluster has similar documents which help to find the related documents in the terms of time and relevancy.

4. Example

In this example there are six web documents and four keywords as shown in figure 3.1. By applying above algorithm, analyze the data for q= 1, 2.

Firstly, for q=1, there is need to determine the value of δ for equation (i). The largest Euclidean distance between any pair of given data points is 5.39 (between x1 and x7) then δ = 1/5.39 = 0.185

These are data points for q =1

x_1 = (0,0) , x_{21} = (1,1) , x_{22} = (1,3), x_3 =(2,2) , x_4 = (3,1), x_5 = (4,0), x_6 = (5,2)

Now calculate membership grade of R for equation (i)

$$R (x_1 , x_2) = 1- 0.185(1^1 + 1^1)^{1/1} = 0.63$$

$$R (x_1 , x_3) = 0.26$$

When determined, relation R may conveniently be represented by the matrix for the following data points

x_1 = (0,0) , x_{21} = (1,1) ,x_3 =(2,2) , x_4 = (3,1), x_5 = (4,0), x_6 = (5,2)

$$\begin{array}{cc} & \begin{array}{cccccc} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} \end{array} \\ R= \begin{array}{c} \mathbf{1} \\ \mathbf{2} \\ \mathbf{3} \\ \mathbf{4} \\ \mathbf{5} \\ \mathbf{6} \end{array} & \begin{pmatrix} 1.0000 & 0.6300 & 0.2600 & 0.2600 & 0.2600 & 0 \\ 0.6300 & 1.0000 & 0.6300 & 0.6300 & 0.2600 & 0.0750 \\ 0.2600 & 0.6300 & 1.0000 & 0.6300 & 0.2600 & 0.4450 \\ 0.2600 & 0.6300 & 0.6300 & 1.0000 & 0.6300 & 0.4450 \\ 0.2600 & 0.2600 & 0.2600 & 0.6300 & 1.0000 & 0.4450 \\ 0 & 0.0750 & 0.4450 & 0.4450 & 0.4450 & 1.0000 \end{pmatrix} \end{array}$$

Similarly calculate the relation for the following data points
x_1 = (0,0) , x_{22} = (1,3) ,x_3 =(2,2) , x_4 = (3,1), x_5 = (4,0), x_6 = (5,2)

$$\begin{array}{cc} & \begin{array}{cccccc} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} \end{array} \\ R= \begin{array}{c} \mathbf{1} \\ \mathbf{2} \\ \mathbf{3} \\ \mathbf{4} \\ \mathbf{5} \\ \mathbf{6} \end{array} & \begin{pmatrix} 1.0000 & 0.6300 & 0.2600 & 0.2600 & 0.2600 & 0 \\ 0.6300 & 1.0000 & 0.6300 & 0.2600 & 0 & 0.0750 \\ 0.2600 & 0.6300 & 1.0000 & 0.6300 & 0.2600 & 0.4450 \\ 0.2600 & 0.2600 & 0.6300 & 1.0000 & 0.6300 & 0.4450 \\ 0.2600 & 0 & 0.2600 & 0.6300 & 1.0000 & 0.4450 \\ 0 & 0.0750 & 0.4450 & 0.4450 & 0.4450 & 1.0000 \end{pmatrix} \end{array}$$

This relation is not max-min transitive; so the transitive closure for given data points is-

x_1 = (0,0) , x_{21} = (1,1) ,x_3 =(2,2) , x_4 = (3,1), x_5 = (4,0), x_6 = (5,2)

$$\begin{array}{cc} & \begin{array}{cccccc} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} \end{array} \\ R_T= \begin{array}{c} \mathbf{1} \\ \mathbf{2} \\ \mathbf{3} \\ \mathbf{4} \\ \mathbf{5} \\ \mathbf{6} \end{array} & \begin{pmatrix} 1.0000 & 0.6300 & 0.6300 & 0.6300 & 0.6300 & 0.4500 \\ 0.6300 & 1.0000 & 0.6300 & 0.6300 & 0.6300 & 0.4500 \\ 0.6300 & 0.6300 & 1.0000 & 0.6300 & 0.6300 & 0.4500 \\ 0.6300 & 0.6300 & 0.6300 & 1.0000 & 0.6300 & 0.4500 \\ 0.6300 & 0.6300 & 0.6300 & 0.6300 & 1.0000 & 0.4500 \\ 0.4500 & 0.4500 & 0.4500 & 0.4500 & 0.4500 & 1.0000 \end{pmatrix} \end{array}$$

Similarly the transitive closure for secondary data points

$$\begin{array}{cc} & \begin{array}{cccccc} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} \end{array} \\ R_T= \begin{array}{c} \mathbf{1} \\ \mathbf{2} \\ \mathbf{3} \\ \mathbf{4} \\ \mathbf{5} \\ \mathbf{6} \end{array} & \begin{pmatrix} 1.0000 & 0.6300 & 0.6300 & 0.6300 & 0.6300 & 0.4500 \\ 0.6300 & 1.0000 & 0.6300 & 0.6300 & 0.6300 & 0.4500 \\ 0.6300 & 0.6300 & 1.0000 & 0.6300 & 0.6300 & 0.4500 \\ 0.6300 & 0.6300 & 0.6300 & 1.0000 & 0.6300 & 0.4500 \\ 0.6300 & 0.6300 & 0.6300 & 0.6300 & 1.0000 & 0.4500 \\ 0.4500 & 0.4500 & 0.4500 & 0.4500 & 0.4500 & 1.0000 \end{pmatrix} \end{array}$$

This relation induces three distinct partitions of its α – cuts for these points x_1 = (0,0) , x_{21} = (1,1) ,x_3 =(2,2) , x_4 = (3,1), x_5 = (4,0), x_6 = (5,2)

α Є [0, 0.45] : { { x_1 , x_{21} ,x_3 , x_4 , x_5, x_6 }}
α Є (0.45, 0.63] : { { x_1 , x_{21} ,x_3 , x_4 , x_5},{x_6 }}
α Є (0.63, 1] : { { x_1 } , {x_{21} },{x_3 }, {x_4 },{ x_5}, { x_6 }}

Similarity for the second set of data points
$\alpha \in [0, 0.45] : \{ \{ x_1 , x_{21} ,x_3 , x_4 , x_5, x_6 \}\}$
$\alpha \in (0.45, 0.63] : \{ \{ x_1 , x_{21} ,x_3 , x_4 , x_5\},\{x_6 \}\}$
$\alpha \in (0.63, 1] : \{ \{ x_1 \} , \{x_{21} \},\{x_3 \}, \{x_4 \},\{ x_5\}, \{ x_6 \}\}$

Repeat the analysis for q = 2, which also corresponds to the Euclidean distance as above.

$R (x_1 , x_4) = 1- 0.185(3^2 + 1^2)^{1/2} = 0.415$

The relation R may conveniently be represented by the matrix for the following data points

$x_1 = (0,0)$, $x_{21} = (1,1)$,$x_3 =(2,2)$, $x_4 = (3,1)$, $x_5 = (4,0)$, $x_6 = (5,2)$

$$R = \begin{array}{c|cccccc} & \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} \\ \hline \mathbf{1} & 1.0000 & 0.7400 & 0.4800 & 0.4200 & 0.2600 & 0.0047 \\ \mathbf{2} & 0.7400 & 1.0000 & 0.7400 & 0.2600 & 0.4200 & 0.2400 \\ \mathbf{3} & 0.4500 & 0.7400 & 1.0000 & 0.7400 & 0.2600 & 0.4500 \\ \mathbf{4} & 0.4200 & 0.2600 & 0.7400 & 1.0000 & 0.7400 & 0.5900 \\ \mathbf{5} & 0.2600 & 0.4200 & 0.2600 & 0.7400 & 1.0000 & 0.4200 \\ \mathbf{6} & 0.4500 & 0.2400 & 0.4500 & 0.5900 & 0.4200 & 1.0000 \end{array}$$

The transitive closure for above data points is

$$R_T = \begin{array}{c|cccccc} & \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} \\ \hline \mathbf{1} & 1.0000 & 0.7400 & 0.7400 & 0.7400 & 0.7400 & 0.5900 \\ \mathbf{2} & 0.7400 & 1.0000 & 0.7400 & 0.7400 & 0.7400 & 0.5900 \\ \mathbf{3} & 0.7400 & 0.7400 & 1.0000 & 0.7400 & 0.7400 & 0.5900 \\ \mathbf{4} & 0.7400 & 0.7400 & 0.7400 & 1.0000 & 0.7400 & 0.5900 \\ \mathbf{5} & 0.7400 & 0.7400 & 0.7400 & 0.7400 & 1.0000 & 0.5900 \\ \mathbf{6} & 0.5900 & 0.5900 & 0.5900 & 0.5900 & 0.5900 & 1.0000 \end{array}$$

Similarly this relation induces three distinct partitions of its α –cuts for these points $x_1 = (0,0)$, $x_{21} = (1,1)$,$x_3 =(2,2)$, $x_4 = (3,1)$, $x_5 = (4,0)$, $x_6 = (5,2)$

$\alpha \in [0, 0.59] : \{ \{ x_1 , x_{21} ,x_3 , x_4 , x_5, x_6 \}\}$
$\alpha \in (0.59, 0.74] : \{ \{ x_1 , x_{21} ,x_3 , x_4 , x_5\},\{x_6 \}\}$
$\alpha \in (0.74, 1] : \{ \{ x_1 \} , \{x_{21} \},\{x_3 \}, \{x_4 \},\{ x_5\}, \{ x_6 \}\}$

5. Results and Snapshots

This result agrees with our visual perception of geometric clusters in the data. This is undoubtedly due to the use of the Euclidean distance. The dendrogram is a graphical representation of the results of hierarchical cluster analysis. This is a tree-like plot where each step of hierarchical clustering is represented as a fusion of two branches of the tree into a single one. The branches represent clusters obtained on each step of hierarchical clustering. The result of above example is described in the form of dendrogram, in snapshots shown in Fig. 5.1 and fig. 5.2

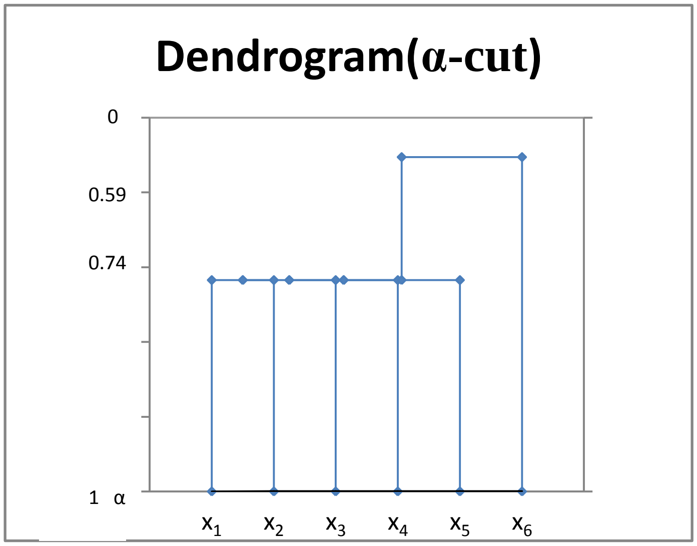

Fig. 5.1: Snapshot of Dendrogram

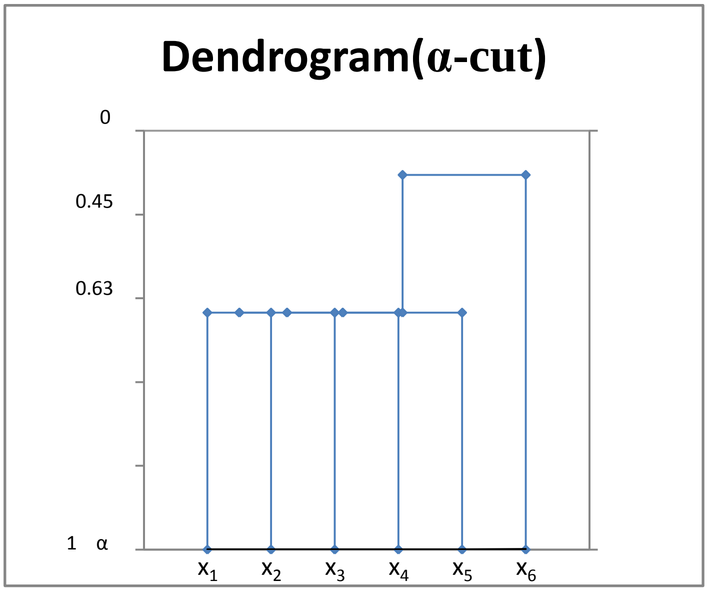

Fig. 5.2: Sanpshot of Dendrogram

The algorithm can effectively find out the points in particular range from a given query point.

6. Conclusion and Future Research

This proposed technique for document retrieval on the web, based on fuzzy logic approach improves relevancy factor. This technique keeps the related documents in the same cluster so that searching of documents becomes more efficient in terms of time complexity.

In future work we can also improve the relevancy factor to retrieval the web documents.

REFERENCES

[1] Agosti, M., Crestani, F., & Pasi, G, “Lectures on information retrieval”. Lecture notes in computer science 1, (2001).

[2]Alltheweb. (2004). Available: http://www.alltheweb.com.

[3] Alsabti, K., Ranka, S., & Singh, V, “An efficient K-means clustering algorithm”, In Proceedings of the 11th international parallel processing symposium (IPPS), (1998).

[4] Baeza-Yates, R., & Ribeiro-Neto, B., “Modern information retrieval”, Addison-Wesley, 1999.

[5] Bailey, P., Craswell, N., & Hawking, D. , “Engineering a multi-purpose test collection for Web retrieval experiments”, Information Processing and Management, 39, 853–871,2003.

[6] Deepti Gupta , Komal Kumar Bhatia, A. K. Sharma “A Novel Indexing Technique For Web Documents Using Hierarchical Clustering”, Vol. 9 No. 9 pp. 168-175 September 30, 2009.

[7] D.R.Cutting, D.R.Karger, J.O.Pedersen And J.W.Tukey. Scatter/Gather: “Cluster-Based Approach To Browsing Large Document Collections”, In Proceedings Of The 15th International Acm Sigir Conference On Research And Development In Information Retrieval, Pages 318-29,1992.

[8] Els Lefever, Timur Fayruzov, Veronique Hoste1, Martine De Cock, “Fuzzy Ants Clustering For Web People Search”, 3 April 20-24, 2009, Madrid, Spain.

[9] Google. (2004). Available: http://www.google.com.

[10] Klir, G., & Yuan, B., “Fuzzy sets and fuzzy logic: theory and applications”. UK: Prentice-Hall, (1995).

[11] P.Willet, “Recent Trends In Hierarchical Document Clustering: A Critical Review”, Information Processing And Management,24: 577-97,1988.

[12] Torra , Sadaaki Miyamoto , Sergi Lanau, “Exploration Of Textual Document Archives Using A Fuzzy Hierarchical Clustering Algorithm In The Gambal System”,Information Processing And Management 41 (2005) 587–598.

[13] Torra, V., Lanau, S., & Miyamoto, S., “Fuzzy clustering for indexing in the GAMBAL information retrieval system”, In Proceedings of the EUSFLAT 2003 (pp. 54–58). Zittau, Germany.

[14] Torra, V., & Miyamoto, S., “Hierarchical Spherical Clustering” , International Journal of Uncertainty Fuzziness and Knowledge-Based Systems, 10(2), 157–172.

[15] Torra, V., & Miyamoto, S., “On increasing the performance of spherical Sammon_s mapping”, In Proceedings of the 7th meeting of the EURO working group on fuzzy sets, workshop on information systems (pp. 17–22).